Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2012, Article ID 818054, 6 pages
doi:10.1155/2012/818054

Review Article

Security Issues in Mobile Ad Hoc Networks

A. L. Sandoval Orozco,[1] J. García Matesanz,[2] L. J. García Villalba,[1] J. D. Márquez Díaz,[3] and T.-H. Kim[4]

[1] *Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA), Facultad de Informática, Universidad Complutense de Madrid (UCM), Despacho 431, Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain*
[2] *Grupo de Análisis, Seguridad y Sistemas (GASS), Sección Departamental de Sistemas Informáticos y Computación, Lenguajes y Sistemas Informáticos y Ciencias de la Computación e Inteligencia Artificial-Facultad de Ciencias Matemáticas, Universidad Complutense de Madrid (UCM), Despacho 310-F, Plaza de Ciencias, 3, Ciudad Universitaria, 28040 Madrid, Spain*
[3] *Grupo de Redes de Computadores e Ingeniería de Software (GRECIS), Departamento de Ingeniería de Sistemas, Universidad del Norte, Km 5 Autopista a Puerto Colombia, Barranquilla, Colombia*
[4] *School of Information Science, GVSA and UTAS, 20 Virginia Court, Sandy Bay, Hobart, TAS 7001, Australia*

Correspondence should be addressed to T.-H. Kim, taihoonn@daum.net

Received 30 September 2012; Accepted 6 October 2012

Academic Editor: Sabah Mohammed

Ad hoc networks are built on the basis of a communication without infrastructure and major investigations have focused on the routing and autoconfiguration problems. However, there is a little progress in solving the secure autoconfiguration problems in mobile ad hoc networks (MANETs), which has led to the proliferation of threats given the vulnerabilities of MANETs. It is clear that ad hoc networks have no centralized mechanism for defense against threats, such as a firewall, an intrusion detection system, or a proxy. Therefore, it is necessary that the defense of interests of each of the ad hoc components is the responsibility of each member node. This paper shows the most common threats to ad hoc networks and reviews several proposals that attempt to minimize some of these threats, showing their protection ability and vulnerabilities in light of the threats that might arise.

1. Introduction

MANET technology is used to immediately provide secure access between multiple mobile nodes without the need for a preset communications infrastructure achieving a multihop architecture. These networks are identified by two basic principles: routing and autoconfiguration.

While there is already quite a lot of established work undertaken on routing [1–4] and consequently those related to secure routing [5–8], there is still a room for continuous improvements on those which are still under construction, notably those related to auto-configuration and in particular, those in connection with secure MANET auto-configuration. Thus, this paper shows the most important works carried out concerning the latter.

Insertion of a node to the MANET involves implementing initial configuration mechanisms [9, 10], such as assigning an available IP address before this node can participate actively in the network. There are three types of solutions to carry out this assignment: *stateful*, *stateless* or *hybrid*.

In *stateful* solutions, addresses are assigned by the network; therefore the network should maintain the status information of addresses that have been assigned and/or released.

In *stateless* solutions, the addresses are assigned by the same node that enters the MANET. This node should run a test for duplicate address detection (DAD) in order to determine the uniqueness of the assigned address.

The *hybrid* solutions combine aspects of both previous types of solutions to improve the scalability and reliability of auto-configuration mechanisms.

All proposals have advantages and disadvantages in terms of solving the following problems: uniqueness of addresses, network initialization, node departure, network partitioning and network merging. However, all lack a mechanism to ensure the authenticity of the address owner at the time in which the auto-configuration is carried out. As a result, a malicious node can spoof any node already set up to hijack its traffic, preventing other nodes from entering the network, sending messages with false addresses, causing denial of service by flooding the network with unsolicited messages from fake addresses,rejecting the possibility that other nodes can access the network, or causing the refusal to accept the insertion of a new node, when the auto-configuration mechanism requires that all nodes confirm the entry of a new member to the MANET.

Although studies over the authenticity of the nodes entering the MANET during auto-configuration have been minimal, the aim of this paper is to show how they have presented some solutions to this problem and show some of its shortcomings from the perspective of the characteristics to be evaluated for potential threats within the auto-configuration process.

This piece of work, including the introduction, is organized into four sections as follows. Section 2 shows an overview of possible threats that may occur within a MANET. In Section 3, the highlights of some proposed solutions to secure MANET auto-configuration are reviewed and analyzed. Finally, conclusions are presented in Section 4.

2. Threats in Autoconfiguration

In the processes applied during the execution of the mechanisms of auto-configuration, predictable and reliable behaviour from the nodes that compose the MANET is expected, as much from those which enter as from those already inside. However, this is not always the case, as malicious nodes can potentially be causing some damage, such as interference of messages, node impersonation, denial of service, spoofing, and eavesdropping among others.

In this paper we use the classification proposed by Wang et al. [11] and Buiati et al. [12] to specify the security threats.

(i) *Address Spoofing Threat*. A malicious node may deliberately choose an assigned or a free IP address for their attack. In the first case, the malicious node teases any configured node as its victim and hijacks its traffic, and in the second one, the node assigns the free IP address to itself to participate in the network, gathering important information necessary to execute active attacks, such as denial of service.

(ii) *Address Space Exhaustion Threat*. A malicious node can claim as many IP addresses as possible until exhausting the address space. This node may request the assignment of addresses to a ghost node (fake nodes). This way the malicious node could prevent other nodes from being configured and entering into the MANET.

(iii) *Address Conflict Threat*. A malicious node can assign a duplicate address to a requester from a possible set of addresses already in use. Thus, it will create, in the DAD process, a blackhole attack of address reply messages (AREP) and lead to an address conflict in the MANET.

(iv) *False Address Conflict Threat*. A malicious node might answer in an unscrupulous way, during the DAD process, using messages AREQ (address request) with false addresses in messages AREP (address reply) that cause conflict with the requester node. Since the victim nodes cannot verify the authenticity of the proposed address, it would have to give up their address and find a new one. The malicious node may change its IP address to execute its attack.

(v) *Denial of Service Threat*. A malicious node could, in an autoconfiguration process, act as a requester and send AREQ messages to multiple initiator nodes simultaneously. Similarly, a malicious node may send many fake DAD messages, causing an overload of traffic.

(vi) *Sybil (Multiples Identities) Threat*. A node illegally claims multiple identities (Sybil node). This node can build a new identity or steal an existing legal node. In general, a Sybil node could demand or assign itself many IP addresses.

(vii) *Negative Reply Threat*. When assigning a new IP address, the approval of all preconfigured nodes is required and an attacker can send a negative response to avoid the entry of the new node.

3. Secure Autoconfiguration

The following are currently the most significant proposals that include secure IP address auto-configuration. The operation of each protocol and what threats they are capable of preventing are explained.

Wang et al. [11] propose a scheme of secure IP address auto-configuration for MANETs, which binds each IP address with a public key allowing each node to authenticate itself into the network and thus prevent spoofing identity and other attacks. The following are considered as the four main security threats surrounding MANET auto-configuration: *address spoofing*, *false address conflict*, *address space exhaustion* and *negative reply threats*.

Identity authentication tries to avoid these threats and this paper proposes to relate every IP address to a public key by means of a one-way hash operation;therefore the owner node of a IP address must use the correspondent key public in order to be authenticated by the network of a unilateral way.

It initiates from the following assumption: the MANET is a network with completely private IP addresses. Therefore, all 32 bits (IPv4) or 131 bits (IPv6) can be used to address nodes in the MANET.

In general, in the proposed scheme, node A, which wants to participate in an existing MANET or start a new one, must first randomly generate a key pair (one public and one private) and one secret key. In the second instance, node A calculates a hash of 32 bits for IPv4 or 131 bits for IPv6.

After calculating the hash value, the node in question temporarily uses the resulting value as its IP address, starts a timer, and broadcasts a *duplicate address probe* (*DAP*) message [13] used to check duplicated addresses on the network.

If a node (node B) configured within the MANET, where Node A wants to enter, finds that the IP address contained in the DAP message issued by node A is equal to it, then it must verify the authenticity of the DAP message. First, node B must check that the IP is equal to the resulting hash of the received public key. Secondly, node B verifies the signature of node A, if it finds that such a signature is correct, then node B checks if public key of node A is equal to it and finally verifies the decryption function. If at least one of the last two checks is not fulfilled, it can be confirmed that there has been an address spoofing attack and therefore node B sends an *address conflict notice* (*ACN*) message via broadcast and discards the received DAP message.

Node A, in turn, waits as long as configured in an internal timer. If it does not receive an ACN message, it assumes that the IP is not in use and permanently assigns the address. If instead it receives an ACN message from some node, before starting the process again, it must verify the authenticity of the ACN message received and the signature of the node issuing the ACN. If these checks are correct, node A is safe that the IP address is assigned to another node and must start the procedure to generate a new pair of public/private keys and secret key; otherwise node A simply discards the ACN message and thus prevents *false address conflict* and *negative reply attacks*.

It is clear that the proposed methodology in the auto-configuration process forces a potential attacker to find, before launching an attack, the public key for which the hash function result is equal to the IP address of the victim, since the controls in the nodes include verification of the identity of the sender node. This process must be applied for each message sent; however the protocol clearly controls address conflict, negative reply, and address spoofing attacks but does not counteract the address exhaustion attack since it does not have a way to specify which node is given which IP addresses, allowing one node to repeat the process as often as desired. This process should be subject to an ACN message which certifies that the node will repeat the process because of IP address duplication.

Buiati et al. [12] propose a secure model for auto-configuration in MANET, based on a distributed and self-organizing certificate system, and also include intrusion detection techniques to improve its safety. The proposed model is built on the protocol DCDP [14] with the improvements proposed by Mohsin and Prakash [15], adopting a collaborative trust model described as "K-out-of-N." So when a new node wants to enter the network, it must earn the trust of K of the N total nodes in the network in order to be accepted into it. To this end, nodes are able to generate certificates with varying degrees of trust. Thus, a distrusted node in the network cannot attack by requesting multiple IP addresses to exhaust them or respond to configuration requests in a malicious manner, as well as allow the implementation of intrusion detection techniques [16].

For the security model, an adversary is defined as any node that produces messages with incorrect auto-configuration protocol information. It then specifies that an adversary can attack the network in two ways: request attacks, where the adversary creates a great number of anomalous messages requesting auto-configuration services, or server attack, where the attacker responds maliciously to requests made by other nodes in the network. In order to avoid these types of attacks, the authors differentiate between trusted and distrusted nodes, avoiding the participation of the latter in the auto-configuration protocol.

Even though there is the possibility that a trusted node is compromised the ability to detect reliable nodes that begin to behave abnormally must be implemented as well. This means that the auto-configuration protocol messages must be authenticated so that an adversary cannot create messages on behalf of another node in the network, being capable of detecting and accusing the adversary nodes. In addition, this detection and accusation system should be implemented collaboratively to prevent an adversary of accusing correct nodes of the network, using the same model "K-out-of-N" explained above.

Authentication of auto-configuration protocol messages is performed using digital signatures, which are built based on digital certificates generated by a distributed certifying authority. This is where the model "K-out-of-N" is applied directly, since, even though every one of the nodes can perform the functions of certifying entity, the entity's private key is split between any subset of K nodes in the MANET. When a new node (one that has not been previously connected to the network) wants to get a digital certificate to identify itself to the MANET, it must take a temporary IP address to request a digital certificate to his 1-hop neighbours. When the MANET nodes receive this request, they can issue a partly signed certificate, depending on the policies established, and send it to the requesting node. After receiving K different certificates, the new node has the ability to build a full certificate and begin the auto-configuration process, discarding the temporary IP. The use of a temporary IP can cause collision problems if the IP is already in use in the network, but it is proposed to use a range of dedicated IPs for this purpose.

The biggest problem in the proposed model is the value of K. A high K value increases security, but reduces the availability of the system because members are less likely to find enough nodes to retrieve the necessary key to the CA. Conversely, if K is small, the availability of the auto-configuration service increases, but the system becomes more vulnerable to attacks by adversaries.

Cavalli and Orset [17] propose a secure auto-configuration protocol adapted to the performance of ad hoc networks, which includes the authentication of the nodes within the network that they will be participating in.

In general, it is intended to satisfy the following items with their secure auto-configuration protocol.

(i) At any time a node must be able to enter or leave the network quickly. Likewise, the network must be able to securely and quickly deliver an IP address to a new node. On the other hand, the abrupt departure of a node must not cause chaos within the network.

(ii) To avoid duplicate IP address conflicts, the protocol must ensure that under no circumstances a node enters the network with its own IP address, but instead the network must be able to deliver the right address to join the MANET.

(iii) The protocol should allow each node to check the veracity of the members of the network to which they belong.

(iv) The protocol should be extremely careful with denial of service. For example, it must not allow a malicious node from monopolizing all IP addresses on the network.

The protocol, in addition to satisfying the described requirements above, wants to meet two broad objectives: the first is to provide a mechanism for IP address auto-configuration for nodes belonging to an ad hoc mobile network, optimizing resources such as bandwidth and time, and the second objective is to allow public key exchange between nodes within the network to ensure the authentication.

The proposed protocol ensures safe IP address auto-configuration including the management of public keys for authentication, which allows avoiding the spoofing attack. However one of its greatest failings is that it neither provides nor supports merging networks or prevents malicious behaviour of network participants after these have been authenticated; among these the denial of service attack is worth mentioning since, for example, a malicious node can authenticate n successive times with n different identities in order to exhaust the available addresses; and another form of attack is that the malicious node refuses to authenticate incoming nodes.

According to Hu and Mitchell [18] the problem with auto-configuration protocols is that their behaviour depends on the correct behavior of all nodes involved. Three attacks are then identified. In the first, a malicious node acts as initiator, assigning duplicate addresses to the requester and sending address assignment messages for nodes that do not exist, effectively reducing the number of addresses available for new valid nodes. The second attack consists of a node acting as a requester, by sending requests for address assignment to multiple initiators, collapsing the network due to broadcast messages generated by the latter in search of a valid IP address. For the third attack, a malicious node can respond to all messages generated by an initiator that tries to find an available IP address, denying access of new nodes to the network.

The proposed solution involves the selection of a method to calculate a "trust value" that is just the level of trust from one node to another, which decreases or increases depending on whether the behaviour of a node is malicious or not, respectively. Then, each node must maintain a list of the levels of trust it has for other nodes. It is possible that different nodes can have different trust limits, depending on security policies. In addition, each node must maintain a blacklist, to which it adds the nodes that do not meet the trust limit, in order to ignore all messages sent by them, except to enable it to recalculate the trust values for these nodes.

When a new node joins the network, it broadcasts a message looking for neighbours, including its trust limit. The nodes receiving this message will respond with a message containing a list of nodes that meet this level of trust, so the new node is able to choose a reliable initiator node. For this model to be fulfilled, the number of malicious nodes needs to be less than the number of normal or valid nodes.

In this way, each time a node receives any information from another node in the network, either as part of the initialization of a new node, collision detection or another process of the auto-configuration protocol, the node first calculates the trust value for the node that sent the message. If this value is below the threshold, the node is added to the black list and a message of suspected malicious node is sent. The nodes receiving this message will act in the same way as the first node, and if they find that the node that sent the message of suspected malicious node has a sufficient trust level, the trust value will be calculated for the suspected node, thus ensuring that only reliable nodes are part of the network. Hu and Mitchell [18] propose a process for calculating the trust level and mention other methods [19, 20].

In the analysis of the trust model, only nodes that consistently behave maliciously are noted. That is, those malicious nodes whose only interest is to affect the calculation of trust values of other nodes are not taken into account and they remain as a weakness in the proposal. Other weaknesses in the proposal are caused by the lack of guarantees against Sybil attacks, where a node uses multiple identities in a fraudulent manner, and against identity theft attacks.

Taghiloo et al. [21] propose the Virtual Address Space Mapping protocol (VASM), where nodes are classified into four categories.

(i) *Allocator*: maintain the address space. They assign new addresses to nodes that join the network.

(ii) *Initiator*: intermediate nodes between the Allocator and the Requester node that exchange all messages between them.

(iii) *Requester*: a new node that needs to get an IP address in order to join the network.

(iv) *Normal*: all the other nodes.

According to this protocol, when a new node joins the ad hoc network, it sends a single hop message called *INITIATOR_SEARCH* to find an Initiator node. If there is no response for this message, the node assumes that it is the only node in the network and begins the network creation process. If the new node gets more than one answer, it selects the sender of the first packet that arrives as an Initiator and sends it an address request. The main task of the Initiator is to get a new IP address from its Allocator and assign it to the requesting node.

In this protocol, each network has at least one Allocator. Each Allocator contains an address space used to assign unique addresses to new nodes as added. The method by which nodes are chosen as an Allocator and how the address space is assigned are the main tasks of the protocol. Similarly, to generate a unique IP address, one Allocator can create another Allocator on the network to balance traffic loads. Each Allocator has a list of all Allocators defined in the network.

The security mechanism for auto-configuration [22] is based on an approach of zero knowledge. This approach only requires a one-way hash function and a seed value, which can be generated randomly. The proposal first establishes a connection between two nodes A and B to exchange information using a cryptographic function on a one-way hash function applied on the seed and in conjunction with a large random number and a secret cryptographic key known by both nodes. Each of the protagonists of the communication carries out the cryptographic operation only the first time and sends the result of the operation and the random number to the other, thus avoiding a man-in-the-middle attack.

For subsequent authentications of both nodes, the value of the seed is increased by one at a time, and the hash function value is calculated on the original value of the seed and the new value increased. The value returned by applying the hash function is sent to the node pair that is being communicated. A node applies the hash function. If the value obtained is equal to the value received the first time, the node is authenticated correctly. For the next communications, the seed value must be incremented by one and the previously explained steps are repeated.

Zhou et al. [23] propose a solution in order to manage the public key of an incoming node, which must be distributed while the secure auto-configuration takes place. Otherwise, a malicious node can impersonate the new node that is registered or that distributes the public key. The SA-PKD achieves the goals of the uniqueness of address allocation and the secure distribution of public key.

It is assumed that the work environment is a densely connected MANET with multiple paths between nodes. If there are malicious nodes in the path between the new node and each of the members, the proposed scheme uses multi-hop broadcast to distribute the information encrypted and signed. Each node checks the forwarded packets to detect the modification of messages.

When a malicious node is placed between a new node and a member of the MANET, it is assumed that there is another good node as a neighbour, and if the malicious node modifies the control message, this node can move or increase the transmission power, sending the message again to try to reach the nodes that lie beyond the malicious one.

If the malicious node deletes the control message, the good node will interpret that the malicious node has left the network or moved away. If there is more than one path between the new node and the MANET member, the message can reach its destination through a different path. If there is a single path, the MANET member will not receive the message because the malicious one interposes and deletes it. The proposal uses the HELLO messages in the routing protocols to help the good node identify the malicious behaviour of the attacker, allowing it to move or increase the transmission power to forward the control message.

4. Conclusions

The insertion of new nodes in a MANET during the auto-configuration process can generate new threats due to the instabilities in the behaviour of these kinds of networks, which would create a lack of trust in the transmission of information through them. The current auto-configuration protocols, with the presented vulnerabilities, have not resolved, in their majority, the security problems found during the insertion of new nodes, creating a necessity for proposals that include this last component. However, the research associated to security during auto-configuration of ad hoc networks is a developing field and still needs much work. In this work, a few existing proposals in the field of secure auto-configuration in MANETs are presented, and they were examined against seven of the most common threats that can be found on these kind of networks to determine how secure or vulnerable they are.

Acknowledgments

This work was supported by the Agencia Española de Cooperación Internacional para el Desarrollo (AECID, Spain) through Acción Integrada MAEC-AECID Mediterráneo A1/037528/11. This work was also supported by the Departamento Administrativo de Ciencia, Tecnología e Innovación (COLCIENCIAS, Colombia) through Programa de Recuperación Contingente which funds Project 121545221101 and the Universidad del Norte through the Dirección de Investigaciones, Desarrollo e Innovación (DIDI).

References

[1] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," RFC 3626, Internet Engineering Task Force, 2003.

[2] D. Jonson, Y. Hu, and D. Maltz, "The Dynamic Source Routing Protocol (DSR) for mobile Ad Hoc networks for IPv4," RFC 4728, Internet Engineering Task Force, 2007.

[3] C. E. Perkins, E. M. Belding-Royer, and S. Das, "Ad Hoc On-Demand Distance Vector (AODV) routing," RFC 3561, Internet Engineering Task Force, 2003.

[4] R. Ogier, F. Templin, and M. Lewis, "Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)," RFC 3684, Internet Engineering Task Force, 2004.

[5] Y.-C. Hu, D. B. Johnson, and A. Perrig, "SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks," *Ad Hoc Networks*, vol. 1, no. 1, pp. 175–192, 2003.

[6] S. Gupte and M. Singhal, "Secure routing in mobile wireless ad hoc networks," *Ad Hoc Networks*, vol. 1, no. 1, pp. 151–174, 2003.

[7] H. Deng, W. Li, and D. P. Agrawal, "Routing security in wireless ad hoc networks," *IEEE Communications Magazine*, vol. 40, no. 10, pp. 70–75, 2002.

[8] L. J. García Villalba, J. García Matesanz, D. Rupérez Cañas, and A. L. García Matesanz, "Secure extension to the optimised link state routing protocol," *IET Information Security*, vol. 5, no. 3, pp. 163–169, 2011.

[9] L. J. García Villalba, J. García Matesanz, A. L. S. Orozco, and J. D. Márquez Díaz, "Auto-configuration protocols in mobile ad hoc networks," *Sensors*, vol. 11, no. 4, pp. 3652–3666, 2011.

[10] L. J. García Villalba, J. García Matesanz, A. L. García Matesanz, and J. D. Márquez Díaz, "Distributed Dynamic Host Configuration Protocol (D2HCP)," *Sensors*, vol. 11, no. 4, pp. 4438–4461, 2011.

[11] P. Wang, D. S. Reeves, and P. Ning, "Secure address autoconfiguration for mobile ad hoc networks," in *Proceedings of the 2nd Annual International Conference on Mobile and Ubiquitous Systems-Networking and Services (MobiQuitous '05)*, pp. 519–521, July 2005.

[12] F. Buiati, R. Puttini, R. de Sousa Jr., C. J. Barenco Abbas, and L. J. García Villalba, "Authentication and autoconfiguration for MANET nodes," in *Proceedings of the 2nd International Conference on Embedded and Ubiquitous Computing (EUC '04)*, pp. 41–52, 2004.

[13] A. Abdelmalek, M. Feham, and A. Taleb-Ahmed, "On Recent Security Enhancements to Autoconfiguration Protocols for MANETs Real threats and requirements," *International Journal of Computer Science and Network Security*, vol. 9, no. 4, pp. 401–407, 2009.

[14] A. Misra, S. Das, A. McAuley, and S. K. Das, "Autoconfiguration, registration, and mobility management for pervasive computing," *IEEE Personal Communications*, vol. 8, no. 4, pp. 24–31, 2001.

[15] M. Mohsin and R. Prakash, "IP address assignment in a mobile ad hoc network," in *Proceedings of the Global Information GRID—Enabling Transformation through 21st Century Communications (MILCOM '02)*, pp. 856–861, Anaheim, Calif, USA, October 2002.

[16] R. S. Puttini, J. Marc Percher, L. M. Mé et al., "A Modular Architecture for Distributed IDS in MANET," in *Proceedings of the International Conference on Computational Science and Its Applications: Part III*, pp. 91–113, Montreal, Canada, May 2003.

[17] A. Cavalli and J.-M. Orset, "Secure hosts auto-configuration in mobile Ad hoc networks," in *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops*, pp. 809–814, Hachioji, Tokyo, March 2004.

[18] S. Hu and C. Mitchell, "Improving IP address autoconfiguration security in manets using trust modelling," in *Proceedings of the 1st International Conference on Mobile Ad-Hoc and Sensor Networks*, vol. 3794 of *Lecture Notes in Computer Science (LNCS)*, pp. 83–92, Wuhan, China, December 2005.

[19] C. Huang, H. P. Hu, and Z. Wang, "Modeling time-related trust," in *Proceedings of the Grid and Cooperative Computing Workshops*, vol. 3252 of *Lecture Notes in Computer Science (LNCS)*, pp. 382–389, October 2004.

[20] A. A. Pirzada and C. McDonald, "Establishing trust in pure Ad-Hoc networks," in *Proceedings of the 27th Australasian Conference on Computer Science*, pp. 47–54, Dunedin, New Zealand, 2004.

[21] M. Taghiloo, M. Dehghan, J. Taghiloo, and M. Fazio, "New approach for address auto-configuration in MANET based on virtual address space mapping (VASM)," in *Proceedings of the 3rd International Conference on Information and Communication Technologies: From Theory to Applications (ICTTA '08)*, pp. 1–6, Damascus, Syria, April 2008.

[22] M. Tajamolian, M. Taghiloo, and M. Tajamolian, "Lightweight secure IP address auto-configuration based on VASM," in *Proceedings of the International Conference on Advanced Information Networking and Applications Workshops (WAINA '09)*, pp. 176–180, May 2009.

[23] H. Zhou, M. W. Mutak, and L. M. Ni, "Secure autoconfiguration and public-key distribution for mobile ad-hoc networks," in *Proceedings of the IEEE 6th International Conference on Mobile Adhoc and Sensor Systems (MASS '09)*, pp. 256–263, Macau, China, October 2009.